REMARKS

Claims 2 to 10, 55 to 63, and 114 to 117 are pending in the present application.

Claims 2-10, 55-63 and 114-117 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over "MatLab Primer" (hereinafter "MatLab"). These rejections are respectfully traversed by the following remarks.

Claim 5 of the present application recites:

> 5. A method for restoring a previous version of a three dimensional mesh model on a computer system comprising:
>
> retrieving a stored copy of an earlier state of the three dimensional mesh model on the computer system;
>
> retrieving an ordered list of operations on the computer system; and
>
> performing at least some of the operations in the ordered list of operations on the retrieved copy of the three dimensional mesh model;
>
> wherein the ordered list of operations contains the operations which if performed in order on the earlier state of the three dimensional mesh model would result in a current state of the three dimensional mesh model.

Claim 58 of the present application recites:

> 58. An article of manufacture comprising a computer-readable medium having stored thereon instructions adapted to be executed by a processor, the instructions which, when executed, define a series of steps to be used for restoring a previous version of a three dimensional mesh model, said steps comprising:
>
> retrieving a stored copy of an earlier state of the three dimensional mesh model;
>
> retrieving an ordered list of operations; and
>
> performing at least some of the operations in the ordered list of operations on the retrieved copy of the three dimensional mesh model;
>
> wherein the ordered list of operations contains the operations which if performed in order on the earlier state of the three dimensional mesh model would result in a current state of the three dimensional mesh model.

Claims 5 and 58 recite retrieving a stored copy of an earlier state of a mesh model and an ordered list of operations and performing at least some of those operations on the retrieved mesh model. The ordered list of operations contains the operations necessary to transform the earlier state of the mesh model into the current state. Thus, one possible application of the example embodiments of the present invention is an undo function where each retrieved operation may be applied to the retrieved copy of the mesh model until the desired "level" of undo is reached. *See also* page 8 of the specification.

The MatLab reference cited by the Examiner describes a programmable environment that allows a user to save the state of the variables in a MatLab session before exiting, to restore the state of these variables later, to create "M-files" that contain sequences of MatLab commands that are performed when the M-file is called, and to produce mesh surface plots and other graphical representations. However, the MatLab reference does not describe the invention of claims 5 and 58 because it does not describe using the capabilities of MatLab to perform the steps recited in those claims. Among other things, the MatLab reference does not describe an "ordered list of operations [that] contains the operations which if performed in order on the earlier state of the three dimensional mesh model would result in a current state of the three dimensional mesh model." While it may be possible for a user to create an M-file in MatLab that contained operations which if performed in order on a earlier state of a three dimensional mesh model would result in a current state of the three dimensional mesh model, Applicants respectfully state that there is no description in the MatLab reference of any user having done so.

To overcome this lack of disclosure in MatLab, the Office relies on an "obvious to try" rationale based on *KSR*. As explained by the Examiner at page 6 of the Office Action,

> it would have been obvious to one of ordinary skill in the art at the time the invention was made to apply M files, meshgrid, and 3D mesh plots disclosed by MatLab and to try using MatLab to assert claim 5, and motivated to perform these functions particularly since the user is encouraged to work at the computer while reading the MatLab Primer and freely experiment with the examples and to draw a 3-D mesh graph.

However, none of the sections cited in MatLab encourages a user to try to put together the functions described in MatLab to arrive at Applicant's claimed process.

As stated in MPEP § 2143, when rejecting a claim based on an "Obvious to try"

rationale, the Examiner must articulate the following:

> (1) a finding that at the time of the invention, there had been a recognized problem or need in the art, which may include a design need or market pressure to solve a problem;
>
> (2) a finding that there had been a finite number of identified, predictable potential solutions to the recognized need or problem;
>
> (3) a finding that one of ordinary skill in the art could have pursued the known potential solutions with a reasonable expectation of success; and
>
> (4) whatever additional findings based on the *Graham* factual inquiries may be necessary, in view of the facts of the case under consideration, to explain a conclusion of obviousness.

In making the rejection, the Examiner does not articulate any of the above findings. For example, the Examiner has not presented, as required in element 2), a finding that there had been a finite number of identified, predictable potential solutions to the recognized need or problem, and element 3) that one of ordinary skill in the art could have pursued the known potential solutions with a reasonable expectation of success.

Applicants submit that discrete software functions described in MatLab are not solutions but merely software routines. In the rejection, the Office has merely identified several discrete software functions performed by the MatLab software, and asserts that one of skill in the art could perhaps use the disclosed software functions duplicate some of the steps of Applicant's claimed invention.

The Federal Circuit in discussing the Supreme Court's *KSR* analysis of an "obvious to try," rationale for a 35 U.S.C. ¶ 103(a) rejection explains in *In re Kubin*, 561 F.3d 1351, 1359-1361 (Fed Cir 2009), that impermissible "obvious to try" cases arises in two situations:
1) where the prior art gave no indication of which parameters were critical or no directions as to which of the many possible choices is likely to be successful, and
2) those in which the prior art gives only general guidance as to the particular form of the claimed invention or how to achieve it.

Applicants submit that both of these situations are met by the Examiner's application of the MatLab reference to the present claims. As to the first situation, the MatLab reference gives no indication of which parameters are critical or no directions as to which of the many possible

choices of software functions are likely to be successful to arrive at Applicants' claimed invention. The MatLab reference merely presents a list of the available functions and instructions on how to use only that particular function. There is no discussion of how to arrange the functions to arrive at the claimed invention, program the functions to arrive at the claimed invention, or even whether the functions work together to make the claimed invention.

As for the second situation, as explained by the Examiner at page 8 of the Office Action, MatLab provides the program statements and a suggestion to try them "(try it)" after each. MatLab provides no guidance or even a hint of how to combine the discrete functions to arrive at the claimed embodiments or how the claimed embodiments may be achieved. Applicants submit the MatLab reference does not provide even general guidance.

As stated in *Kubin*, "the prior art must give direction as to which of the many possible choices is likely to be successful, or provide more than general guidance as to the particular form of the invention or how to achieve it." 561 F.3d at 1361.

Applicants respectfully submit that the MatLab reference does not rise to the level of disclosure to meet the "obvious to try" rationale as stated in the MPEP § 2143 and as held by both the US Supreme Court and the Federal Circuit. Furthermore, the Examiner has not met his burden of finding that there was an identified problem at the time of the invention, there existed a finite number of identified, predictable potential solutions to the recognized need or problem; and that one of ordinary skill in the art could have pursued the known potential solutions with a reasonable expectation of success.

Accordingly, the Examiner has not made a *prima facie* case of obviousness with respect to the features recited in claims 5 and 58. Based on the above remarks, claims 5 and 58 are allowable.

As to the rejections of claims 6-10 and 59-63 as being unpatentable over the MatLab reference, Applicants submit that the Examiner has failed to make a *prima facie* case of obviousness for these claims based on the above remarks with respect to claims 5 and 58. Claims 6-10 and 59-63 depend from claims 5 and 58, respectively, and are also allowable.

As to claims 2 to 4, 55 to 57 and 114 to 117, Applicants respectfully submit that the rejections should be withdrawn as explained below.

Claim 115 of the present application recites:

115. A method for managing a three dimensional mesh model on a computer system, comprising:

storing a copy of a first state of the three dimensional mesh model on the computer system;

performing operations on the three dimensional mesh model, wherein the three dimensional mesh model is in a second state after performing the operations;

storing a record of each of the operations in an ordered list on the computer system; and

reapplying at least some of the operations stored in the ordered list to the stored first state of the three dimensional mesh model, wherein the three dimensional mesh model is in a third state after reapplying the at least some of the operations.

Claim 116 of the present application recites:

116. An article of manufacture comprising a computer-readable medium having stored thereon instructions adapted to be executed by a processor, the instructions which, when executed, define a series of steps to be used for managing a three dimensional mesh model, said steps comprising:

storing a copy of a first state of the three dimensional mesh model;

performing operations on the three dimensional mesh model, wherein the three dimensional mesh model is in a second state after performing the operations;

storing a record of each of the operations in an ordered list; and

reapplying at least some of the operations stored in the ordered list to the stored first state of the three dimensional mesh model, wherein the three dimensional mesh model is in a third state after reapplying the at least some of the operations.

Claim 117 of the present application recites:

117. A system for managing a three dimensional mesh model, the system comprising:

a computer module that stores a copy of a first state of the three dimensional mesh model;

a computer module that performs operations on the three dimensional mesh model, wherein the three dimensional mesh model is in a second state after performing the operations;

a computer module that stores a record of each of the operations in an ordered list; and

> a computer module that *reapplies at least some of the operations stored in the ordered list to the stored first state of the three dimensional mesh model*, wherein the three dimensional mesh model is in a third state after reapplying the at least some of the operations.

Similar to the discussion above in regard to claims 5 and 58, while the MatLab reference describes functionality for storing lists of operations and graphically rendering mesh surface plots, it does not describe the inventions of claims 115, 116 or 117 actually being performed or assembled.

For example, in regard to claim 115, while the MatLab reference describes the MatLab system as being capable of storing list of operations (M-files) and as being capable of rendering mesh surface plots, the MatLab reference does not describe storing a copy of a first state of a three dimensional mesh model, performing operations on the three dimensional mesh model to place it in a second state, storing a record of each of the operations in an ordered list, and reapplying at least some of the operations stored in the ordered list to the stored first state of the three dimensional mesh model to place it in a third state as recited by claim 115. The MatLab reference similarly does not describe storing instructions on a computer readable medium to do so (i.e., as in claim 116), or assembling computer modules to do so (i.e., as in claim 117).

To overcome the above described deficiencies of MatLab, the Examiner relies on an "obvious to try" rationale based on *KSR*. However, none of the sections cited in MatLab encourages a user to try to put together the functions described in MatLab to arrive at Applicant's claimed process. As explained above MPEP § 2143 states that Examiner must articulate several findings. However, in making the rejection, the Examiner does not articulate any of the findings. For example, the Examiner has not presented, as required in § 2143, a finding that there had been a finite number of identified, predictable potential solutions to the recognized need or problem, and a finding that one of ordinary skill in the art could have pursued the known potential solutions with a reasonable expectation of success.

Applicants submit that discrete software functions described in MatLab are not solutions but merely software routines. In the rejection, the Office has merely identified several discrete software functions performed by the MatLab software, and asserts that one of skill in the art could perhaps use the disclosed software functions duplicate some of the steps of Applicant's

claimed invention.

The Federal Circuit in discussing the Supreme Court's *KSR* analysis of an "obvious to try," rationale for a 35 U.S.C. ¶ 103(a) rejection explains in *In re Kubin*, 561 F.3d at 1359-1361, that impermissible "obvious to try" cases arises in two situations:

1) where the prior art gave no indication of which parameters were critical or no directions as to which of the many possible choices is likely to be successful, and

2) those in which the prior art gives only general guidance as to the particular form of the claimed invention or how to achieve it.

Applicants submit that both of these situations are met by the Examiner's application of the MatLab reference to the present claims. As to the first situation, the MatLab reference gives no indication of which parameters are critical or no directions as to which of the many possible choices of software functions are likely to be successful to arrive at Applicants' claimed invention. The MatLab reference merely presents a list of the available functions and instructions on how to use only that particular function. There is no discussion of how to arrange the functions to arrive at the claimed invention, program the functions to arrive at the claimed invention, or even whether the functions work together to make the claimed invention.

As for the second situation, as explained by the Examiner at page 8 of the Office Action, MatLab provides the program statements and a suggestion to try them "(try it)" after each. MatLab provides no guidance or even a hint of how to combine the discrete functions to arrive at the claimed embodiments or how the claimed embodiments may be achieved. Applicants submit the MatLab reference does not provide even general guidance.

As stated in *Kubin*, "the prior art must give direction as to which of the many possible choices is likely to be successful, or provide more than general guidance as to the particular form of the invention or how to achieve it." 561 F.3d at 1361.

Applicants respectfully submit that the MatLab reference does not rise to the level of disclosure to meet the "obvious to try" rationale as stated in the MPEP § 2143 and as held by both the US Supreme Court and the Federal Circuit. Furthermore, the Examiner has not met his burden of finding that there was an identified problem at the time of the invention, there existed a finite number of identified, predictable potential solutions to the recognized need or problem; and that one of ordinary skill in the art could have pursued the known potential solutions with a

reasonable expectation of success.

Accordingly, the Examiner has not made a *prima facie* case of obviousness with respect to the features recited in claims 115 -117. Based on the above remarks, claims 115-117 are allowable.

Claims 2 to 4, 55 to 57 and 114 depend from claims 115, 116 and 117. Accordingly, the arguments presented above in connection with claims 115, 116 and 117 apply equally to claims 2 to 4, 55 to 57 and 114. In view of the foregoing, it is respectfully submitted that the rejection of claims 2 to 4, 55 to 57, 114, 115, 116 and 117 under 35 U.S.C. § 103 in view of the MatLab reference should be withdrawn.

Applicants respectfully submit that all pending claims are in condition for allowance. Prompt consideration and allowance of the present application are therefore earnestly solicited.

The Office is hereby authorized to charge any additional fees or credit any overpayments under 37 C.F.R. § 1.16 or § 1.17 to Deposit Account No. 11-0600.

The Examiner is invited to contact the undersigned at (212) 425-7200 to discuss the application.

Respectfully submitted,

Dated: October 30, 2009

By: /Martin E. Miller/

Martin E. Miller (Reg. No. 56,022)
KENYON & KENYON LLP
1500 K Street, NW
Washington, DC 20005
(202) 220-4200 (telephone)
(202) 220-4201 (facsimile)